CCSDS Spacecraft Monitor & Control Service Framework

Mario Merri[1], Michael Schmidt[1], Alessandro Ercolani[1], Ivan Dankiewicz[1], Sam Cooper[1]
Roger Thompson[2], Martin Symonds[2]
Amalaye Oyake[3], Ashton Vaughs[3], Peter Shames[3]
Donald Custis Lokerson[4]
Brigitte Béhal[5], Christine Comier[5]
Harald Hofmann[6], Klaus Gnadl[6]
Takahiro Yamada[7]

[1]European Space Agency (ESA)/Europe
[2]*British National Space Centre (BNSC)/United Kingdom*
[3]*National Aeronautics and Space Administration Jet Propulsion Laboratory (NASA/JPL)/USA*
[4]*National Aeronautics and Space Administration Goddard Space Flight Centre (NASA/GSFC)/USA*
[5]*Centre National d'Etudes Spatiales (CNES)/France*
[6]*Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR)/Germany*
[7]*Japan Aerospace Exploration Agency(JAXA)/Japan*

This CCSDS paper presents a reference architecture and service framework for spacecraft monitoring and control. It has been prepared by the Spacecraft Monitoring and Control working group of the CCSDS Mission Operations and Information Management Systems (MOIMS) area.
In this context, Spacecraft Monitoring and Control (SM&C) refers to end-to-end services between on-board or remote applications and ground-based functions responsible for mission operations.
The scope of SM&C includes:

1) Operational Concept: definition of an operational concept that covers a set of standard operations activities related to the monitoring and control of both ground and space segments.

2) Core Set of Services: definition of an extensible set of services to support the operational concept together with its information model and behaviours. This includes (non exhaustively) ground systems such as Automatic Command and Control, Data Archiving and Retrieval, Flight Dynamics, Mission Planning and Performance Evaluation.

3) Application-layer information: definition of the standard information set to be exchanged for SM&C purposes.

Service Approach

Overview

Service based architecture is gradually replacing monolithic architecture as the main design principle for new applications in both private and distributed systems. It is one of the fundamental design principles of network distributed applications where the interface, both operations and data objects, must be well defined as the clients are often heterogeneous.

The architecture is essentially a design that starts with an interface definition and builds the entire application based around the interfaces, interface semantics (state machines of the interface) and interface calls (operations allowed and data objects passed). Whilst there are no universally recognised standards for service based architecture, the concepts and terminology used are nevertheless well developed and consistent, having evolved in line with their wide utilisation. Comparison shows that these are analogous to the service model conventions used for the ECSS PUS [R1] and SLE [R2].

Layered Service Architecture

All services are not equal. Some services are high level services that provide very specific functionality whereas others are more generic utility services that may be used by several service consumers. Because of this it is useful to create a layered architecture, where high level services can use lower level services if required.

A key factor of a layered approach is that services are not allowed to invoke operations on a service in a higher level. This does not mean that as part of the service it cannot be required that you provide call back information so that data can be returned asynchronously, what it does mean is that a definite dependency hierarchy can be created that will never have circular dependencies. An example of this is the OSI reference network layer model.

The combination of using well defined services and the layered approach means that interoperability between disparate systems is far more achievable than when using bespoke systems. It is required that at some level the two systems agree on what protocol is used but the advantage of the layered service approach is that this can be hidden from the higher level applications in a lower level communications service.

For example, an application can communicate with a backend system regardless of its platform, communication method and location if the application uses a well defined service interface and some translation code. The translation code provides the standard interface to the application and converts it to the platform specific format required for the back end system. This translation process is known as an adapter:

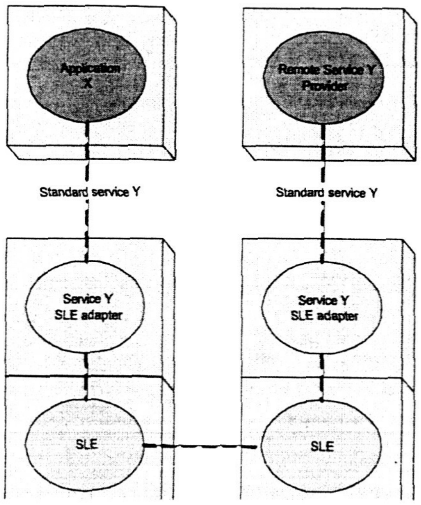


Figure 1: Adapter functions

In the above diagram it is only the adapter that needs to be SLE aware for the application to be interoperable with an SLE based system, i.e. a spacecraft. It is also only the adapter that needs to be changed to support middleware more appropriate for a ground based system, i.e. DCOM, SOAP or CORBA:

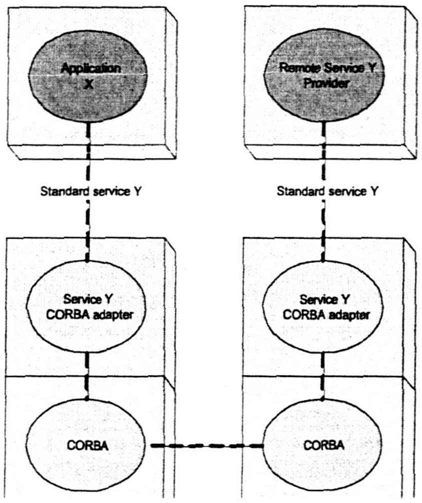


Figure 2: Adapter functions

The service protocol may be operated over different underlying communication protocols depending on the location and characteristics of the communications media used for the link between a service consumer and provider.

Reference Architecture

Before any work on the operational context can be done it is important that a reference architecture be developed. The reference architecture not only shows the physical deployments but also the software deployments too.

To assist in the development and presentation of the reference architecture the CCSDS Reference Architecture for Space Data Systems methodology (RASDS) [3] was used. This model presents five primary views of the system being modelled, two of which are detailed below, and allows these to be combined to present more complex views of the system.

Functional

This view shows the separate areas of functionality that are involved in the operation of a spacecraft. The connections shown between functions are logical ones; they use underlying communication protocols running over actual physical links to transfer data:

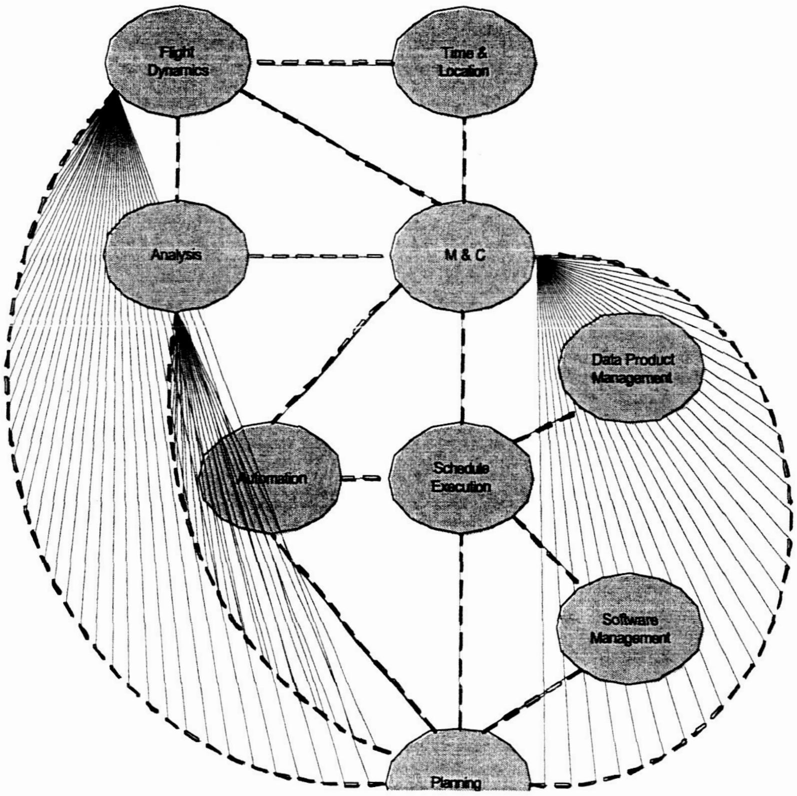


Figure 3: Functional view

The characters of the data items that are exchanged over these logical links is an aspect detailed in the information views.

For example, for an advanced spacecraft these automation functions may be present onboard and the ground based systems would only acquire status data from these autonomous systems in order to monitor and guide their operation, see Figure 4. These guidance systems would use high level goals (e.g. "take an observation of Quasar 3C273", or "drive to that rocky outcropping") rather than low level commands (e.g. turn on power, open shutter, take 1 min exposure, close shutter, dump data, etc) to direct on-board activities:

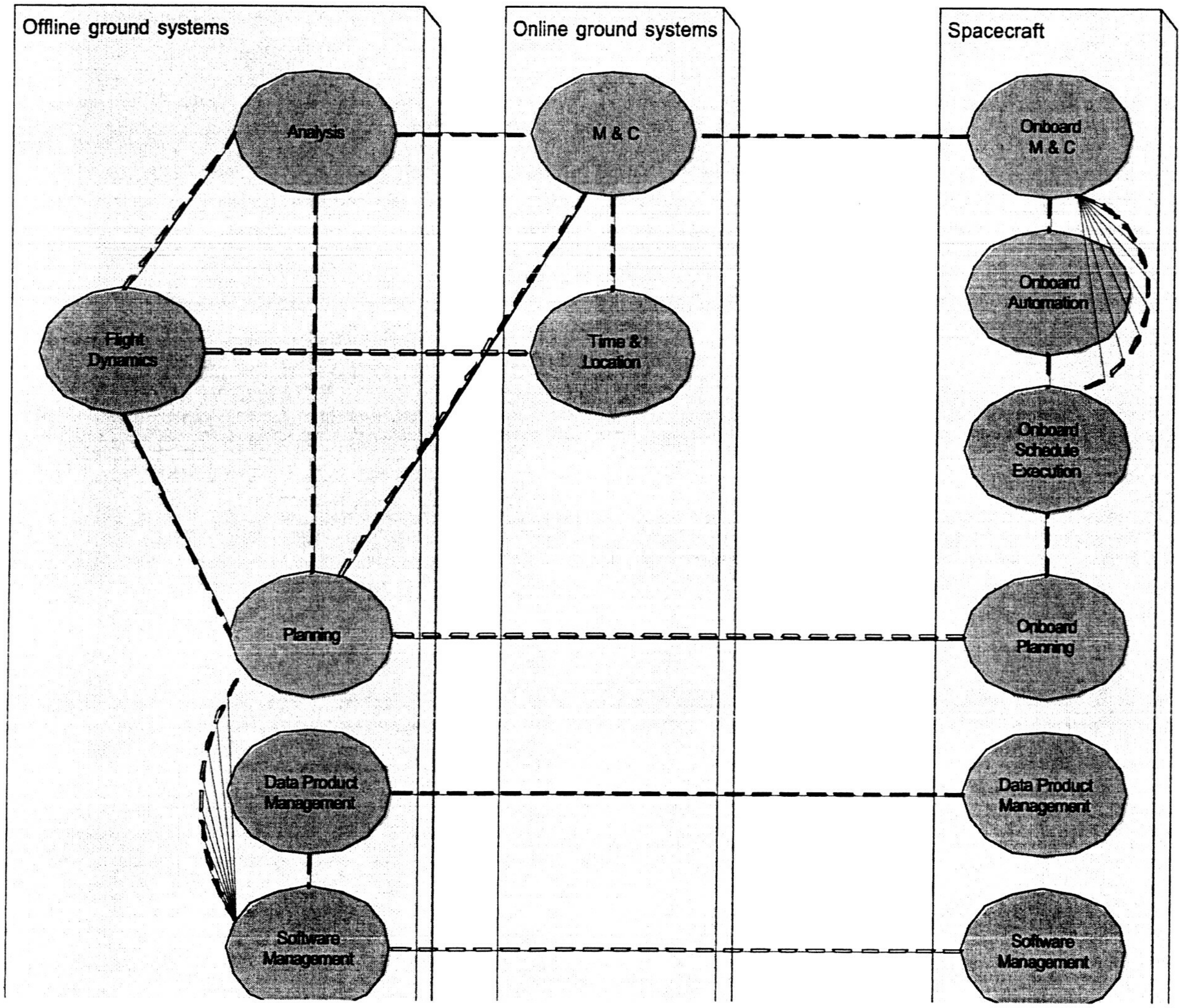


Figure 4: Advanced spacecraft functional example

The above deployment view illustrates that a functional object may be required to be present in more than one location, for example Software Management. This splitting is required when the functionality is required in more than one location, in the case of Software Management it is required in the ground systems for the production of the software update and also in the onboard systems for the applying of the software update.

Communication

It is envisaged that a standard low level service is required by the high level services to transfer the data objects to the destination service provider, whether that is space or ground based. This Common SM&C service hides the underlying communications protocols from the higher level services. Data objects that are likely to be required to be transported are:

- Directive
- Event

Supported operations of the Common SM&C service on the data objects would be:

- Send Directive or Event

The diagram below shows an example of the communication objects being used in a layered way. The high level application communicates through the high level services to the high level service provider on the spacecraft. The high level service protocol uses the common services [Common SM&C and potentially others] to communicate with its peer and uses appropriate network layers to provide this link. Different common service implementations would most likely be required for space-ground and ground-ground interfaces if different communications protocols were used:

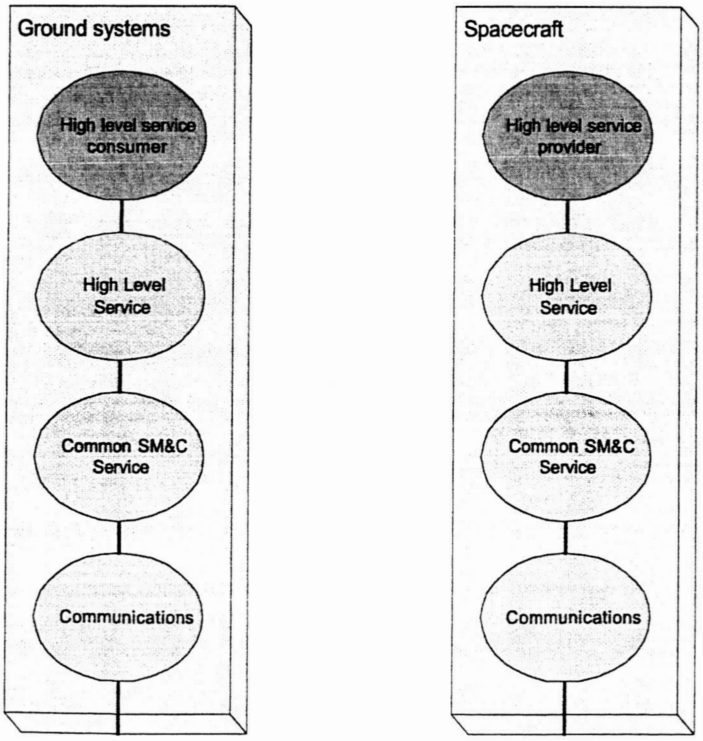


Figure 5: High Level Service Concept Communications View

The definition of a standard interface and protocol allows the low level communication protocol to be hidden from the higher level applications and services. The figure below demonstrates this layer of communication protocols in place:

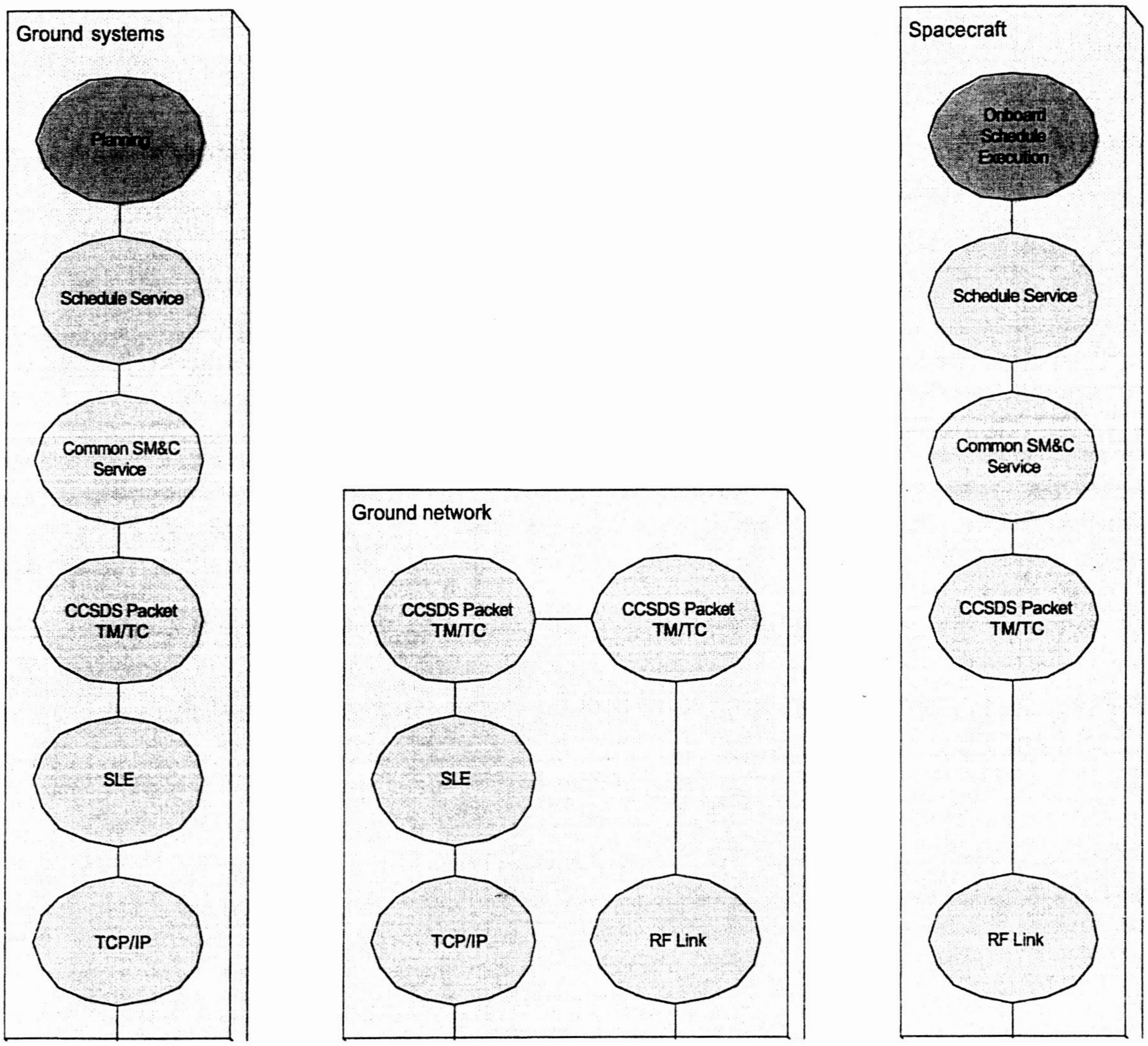


Figure 6: Onboard Schedule Execution Service Communications View

On the ground system node the Schedule Service communication object provides an interface which allows the Planning functional object to communicate with the (Onboard) Schedule Execution functional object. This interface is independent of the lower level communication protocols required to provide the interface. The Schedule Service uses the Common SM&C Service communication object to provide the communication link.

On the spacecraft the (Onboard) Schedule Execution functional object interfaces with other onboard functional objects via relevant onboard communication objects (not shown). Again the interface provided by the communication objects are independent of the lower level communication protocols required to provide the interface.

Operational Context

The operational context was derived from experience with existing system and also by examining existing operational service based architectures. By combining the information from both the operational context and the reference architecture it was possible to derive a list of high level services:

[illegible]	[illegible]
SM&C	Core Monitoring and Control service.
Automation	Activity automation management.
Scheduling	Activity scheduling.
Interaction	Operator notification and interaction.
Planning	Constraint and resource planning.
Flight Dynamics	Orbit determination, flight plan generation, and manoeuvre generation.
Time & Location	Time correlation, tracking, ranging, and onboard position determination services.
Data Product Management	File management and transfer, both ground based and onboard.
Software Management	Software versioning, patching, and release.

This list is based on the identification of a core set of operational functions that exist in many spacecraft control systems, and the services that these offer. These functions may be distributed between organisations and physical entities in many different ways; increasingly, functions normally considered ground based are now being deployed on-board spacecraft. The intention is to identify service interfaces that allow different distributions of operational functions to be adopted by individual organisations or missions.

The figure below illustrates the concept for the definition of a set of high level operational services for the Monitoring and Control of spacecraft missions:

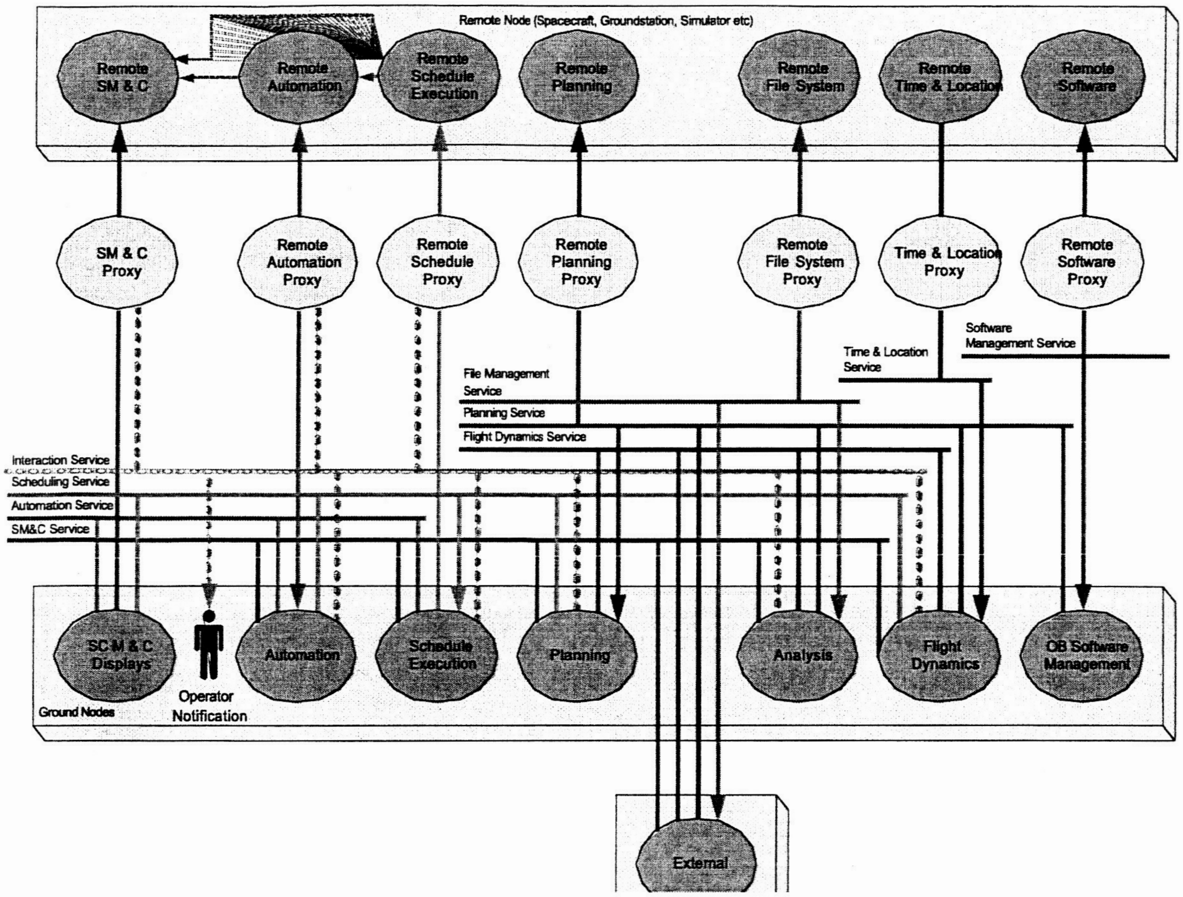


Figure 7: Services Overview

These high-level services relate to end-to-end operation of spacecraft [and potentially other remote systems] and would be independent of underlying transport protocols. Through the use of a layered service architecture these same services may be deployed over a protocol stack that use CCSDS Packet TM/TC for space-ground connections, while being used over TCP/IP to control an equivalent ground-based function. This enables functions to be migrated from ground to space without impacting client functions within the ground segment.

Also legacy systems can be incorporated within the concept through the use of adapter functions that present a service façade to client functions. Proxies can also be used for missions where space-ground contact is intermittent.

Conclusion

The deployment of standard services and interfaces for the provision of high level services can significantly reduce the amount of customisation required for the support of the high level operations of a spacecraft.

The use of adapter components, where one protocol is converted to another, allows hardware manufacturers and software developers to provide interoperability simply and for a lower cost. The adapters become standard software which allows a specific spacecraft platform to be used with any compliant control system as the adapters hide the platform specifics and just expose the standard interface.

Where the corresponding service interface crosses the boundary between operating agencies, missions or systems, then the adoption of standard services offers interoperability between infrastructures. When the corresponding service interface crosses the boundary between software systems, then the adoption of standard services permits the development of "plug and play" components from various organisations. This provides the capability of rapid integration into a mission specific system using underlying communications objects to be selected that are relevant to the environment infrastructure.

[1] *Telemetry and Telecommand Packet Utilisation Standard (PUS)* [ECSS-E-70-41A 30 January 2003]

[2] *Space Link Extension Service Specifications.* [Space Link Extension Service Specifications are in various stages of development. Current issues of CCSDS documents are maintained at the CCSDS Web site: http://www.ccsds.org.]

[3] *Reference Architecture for Space Data Systems (RASDS).* CCSDS Architecture Working Group. Issue 0.8a, 4 November 2003.

Spacecraft Monitor & Control Working Group (SM&C WG)

CCSDS SM&C WG

Membership

- ESA
  - Mario Merri (chairman), Michael Schmidt, Alessandro Ercolani
  - Ivan Dankiewicz, Sam Cooper
- BNSC
  - Roger Thompson (deputy chairman), Martin Symonds
- NASA/JPL
  - Amalaye Oyake, Ashton Vaughs, Peter Shames
- NASA/GSFC
  - Donald Custis Lokerson, Danford. S. Smith (observer)
- CNES
  - Brigitte Béhal, Christine Cornier
- DLR
  - Harald Hofmann, Klaus Gnadl
- JAXA
  - Takahiro Yamada

High Level Goal

- Standardisation of interfaces for Spacecraft M&C
  - Reduced cost of Flight Components and Ground Segment Infrastructure
  - Enable "plug and play" Architecture with components from different Agencies, Systems and Suppliers

- Enable Mission Economies through:
  - Interoperability with partner systems and infrastructure
  - Risk Reduction through re-use of systems and operational concepts: increased reliability
  - Facilitation of Generic Software Infrastructure (On-board and Ground-based)
  - Application of a common M&C Approach throughout all mission phases
  - Application of a common M&C Approach to other domains (Ground Stations, Control and Test Centres, etc.)

Scope

- Operational concept:
  - definition of an operational concept that covers a set of standard operations activities related to the monitoring and control of both ground and space segments.
- Core Set of Services:
  - definition of an extensible set of services to support the operational concept together with its information model and behaviours. This includes (non exhaustively) ground systems such as Automatic Command and Control, Data Archiving and Retrieval, Flight Dynamics, Mission Planning, Automation, and Performance Evaluation.
- Application-layer information:
  - definition of the standard information set to be exchanged for SM&C purposes.

Objectives

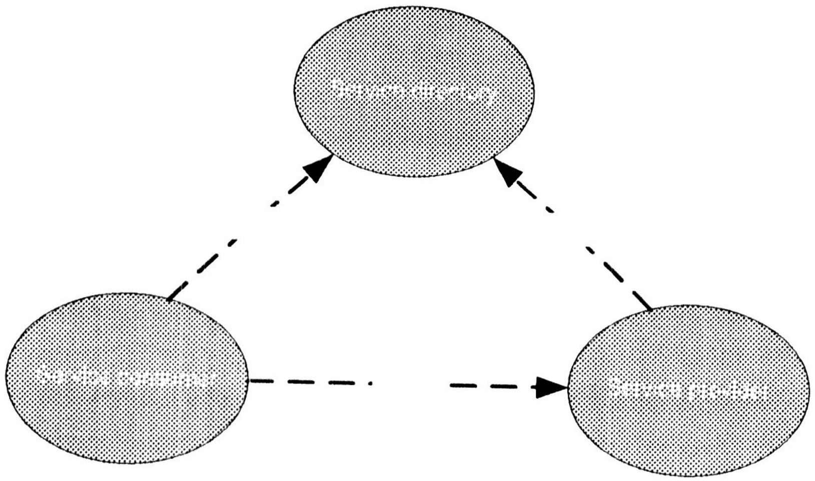
- Define a common SM&C operational concept
- Define a reference on-board and on ground architecture in close coordination with other CCSDS groups (e.g. SOIS, SEA)
- Define a standard approach to service specification
- Define minimum set of ground SM&C kernel functionality
- Define space-to-ground application-level interface messages
- Identify standard Mission Control System interfaces with M&C components:
  - With remote software management
  - With Mission Planning System
  - With external tools requiring monitoring and control data
  - With external users (provision of TM and reception of TC)
  - With the Flight Dynamics System

Schedule

◆ Dec 03	Formation of WG
◆ 12 Jan 04	White Book – draft 0.1
◆ 18 Feb 04	White Book – draft 0.2
◆ 5 Apr 04	White Book – draft 0.3
◆ 30 Apr 04	White Book – draft 0.4
◆ May 04	White Book – Issue 1.0
◆ TBD	Production of Blue books

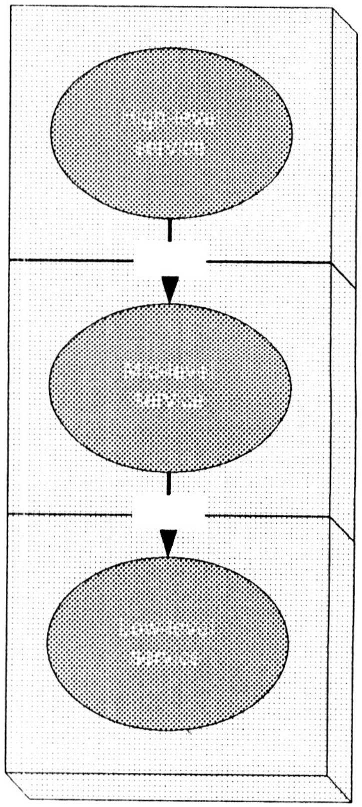
Service Based Architecture

- Defines the interface as a "contract"
- Consumer (client) unaware of providers implementation
- Popular with distributed and web based systems



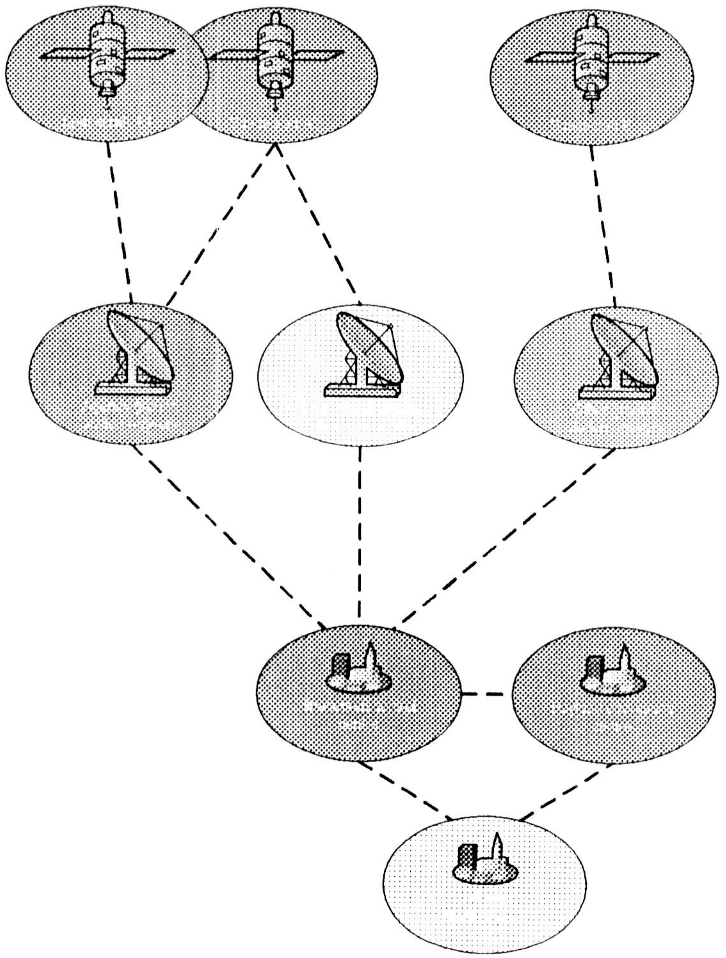
Layered Architecture

- Helps modularise system
- Supports reuse of components
- Can only invoke lower layers
- Replacement of layers is possible if combined with services



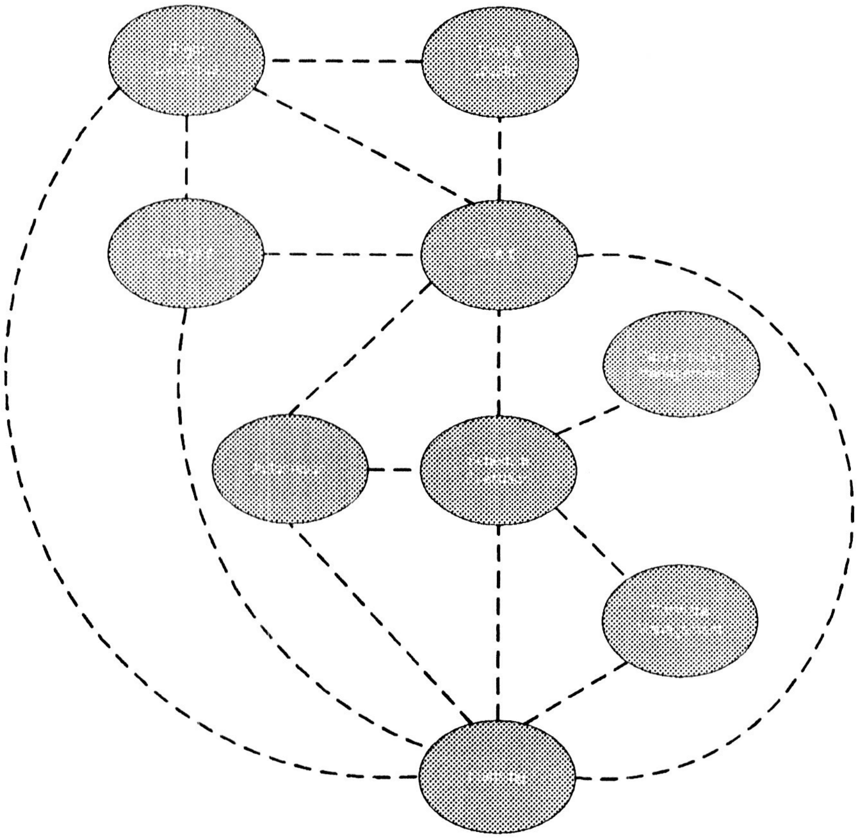
Space System Context

- Defines reference system
- Defines operational context
- ◆ Outlines clients of the system
- ◆ Outlines operational areas of the system



Functional View

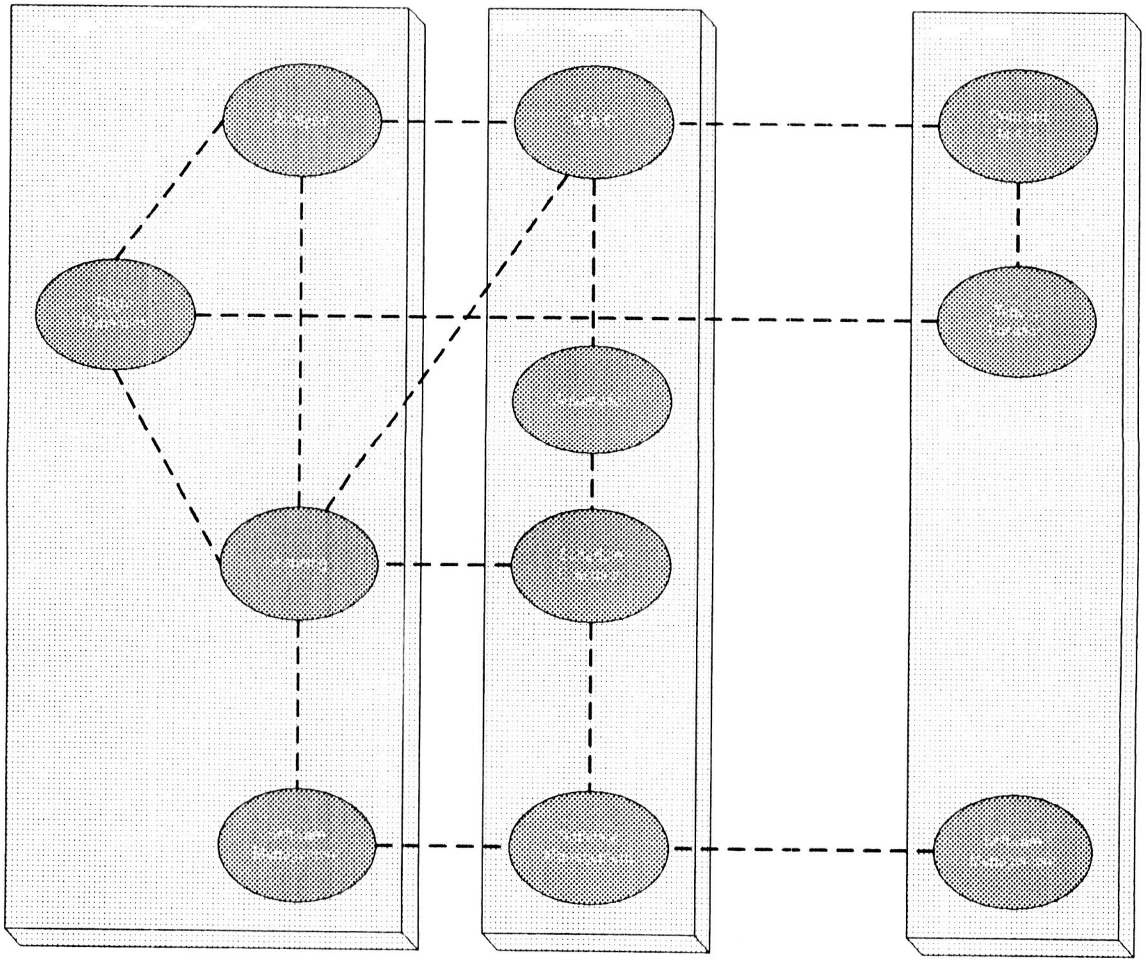
- Uses RASDS methodology
- Defines areas of functionality
  - Taken from the operational context
- Defines interactions between functional areas
- Leads to service identification



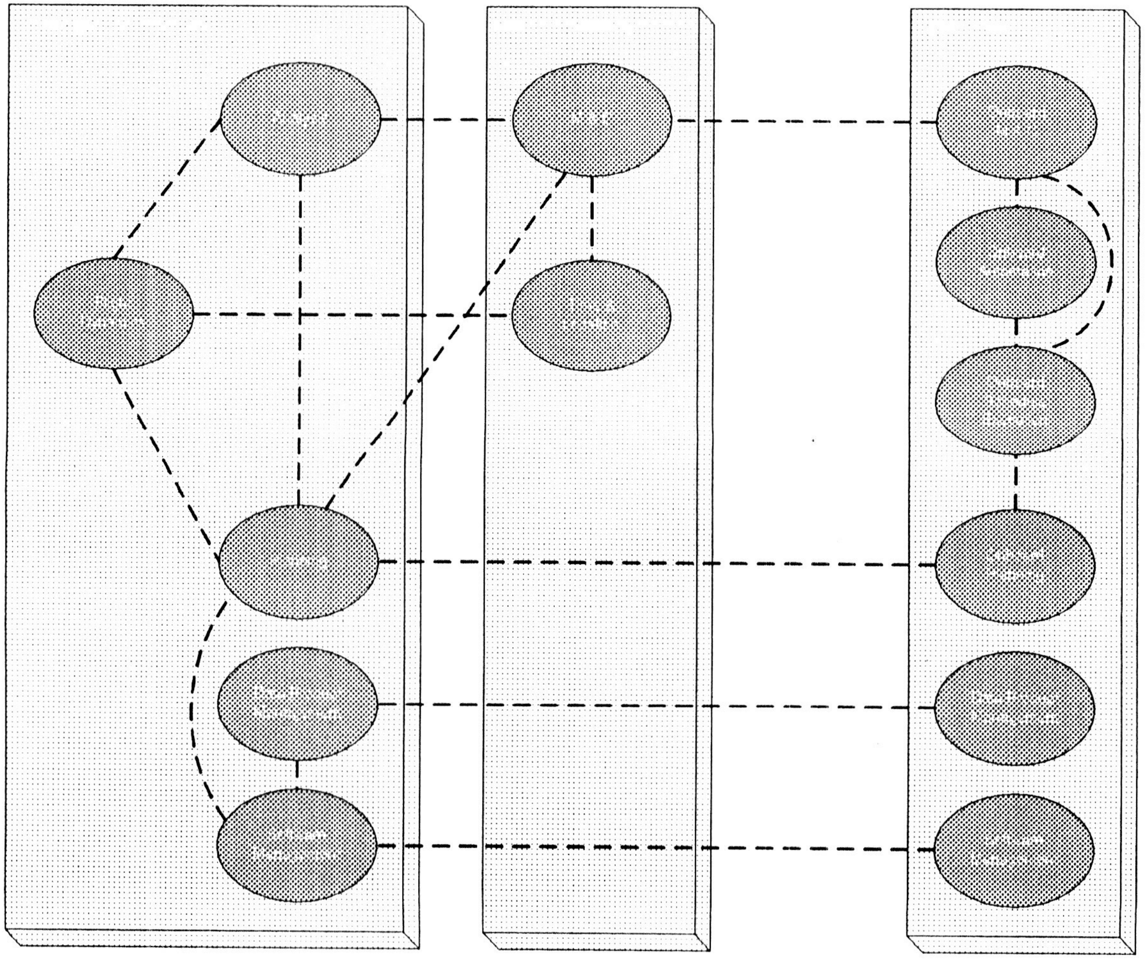
List of Functional Areas

- M&C
  - Core Monitor and Control functional area.
- Automation
  - Concerned with control and management of the automation task.
- Schedule Execution
  - Component responsible for executing the schedule possibly generated by the planning component.
- Planning
  - Component responsible for the planning of future events in the system.
- Software Management
  - Onboard and also ground software version control, patching and release.
- Flight Dynamics
  - Orbit vector determination, event schedule production etc.
- Time
  - Time correlation
- Location
  - Responsibilities include ranging, tracking, and onboard position determination.
- Analysis
  - Trending and analysis.
- Data Product Management
  - Control, management, persistence and transfer of the data product produced by the other components.

Example: Simple craft

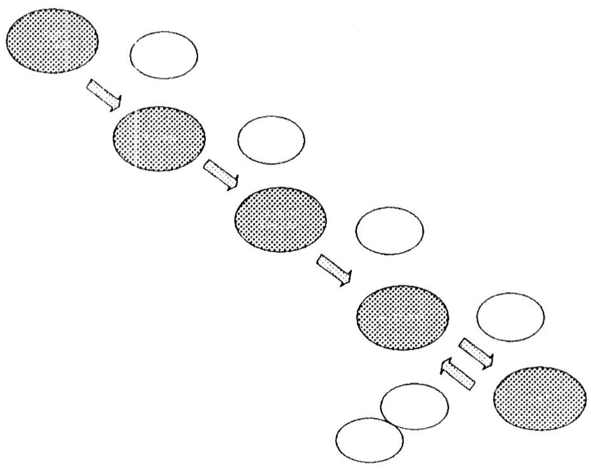


Example: Advanced craft

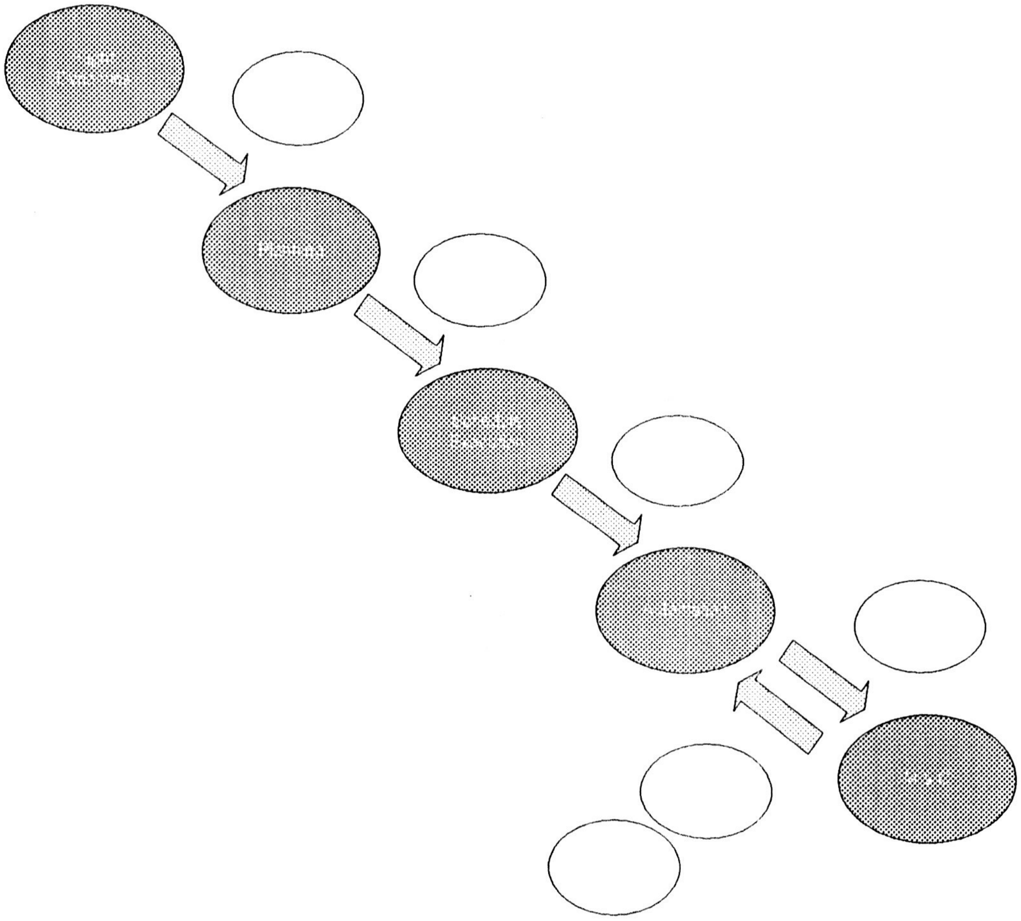


Information flow

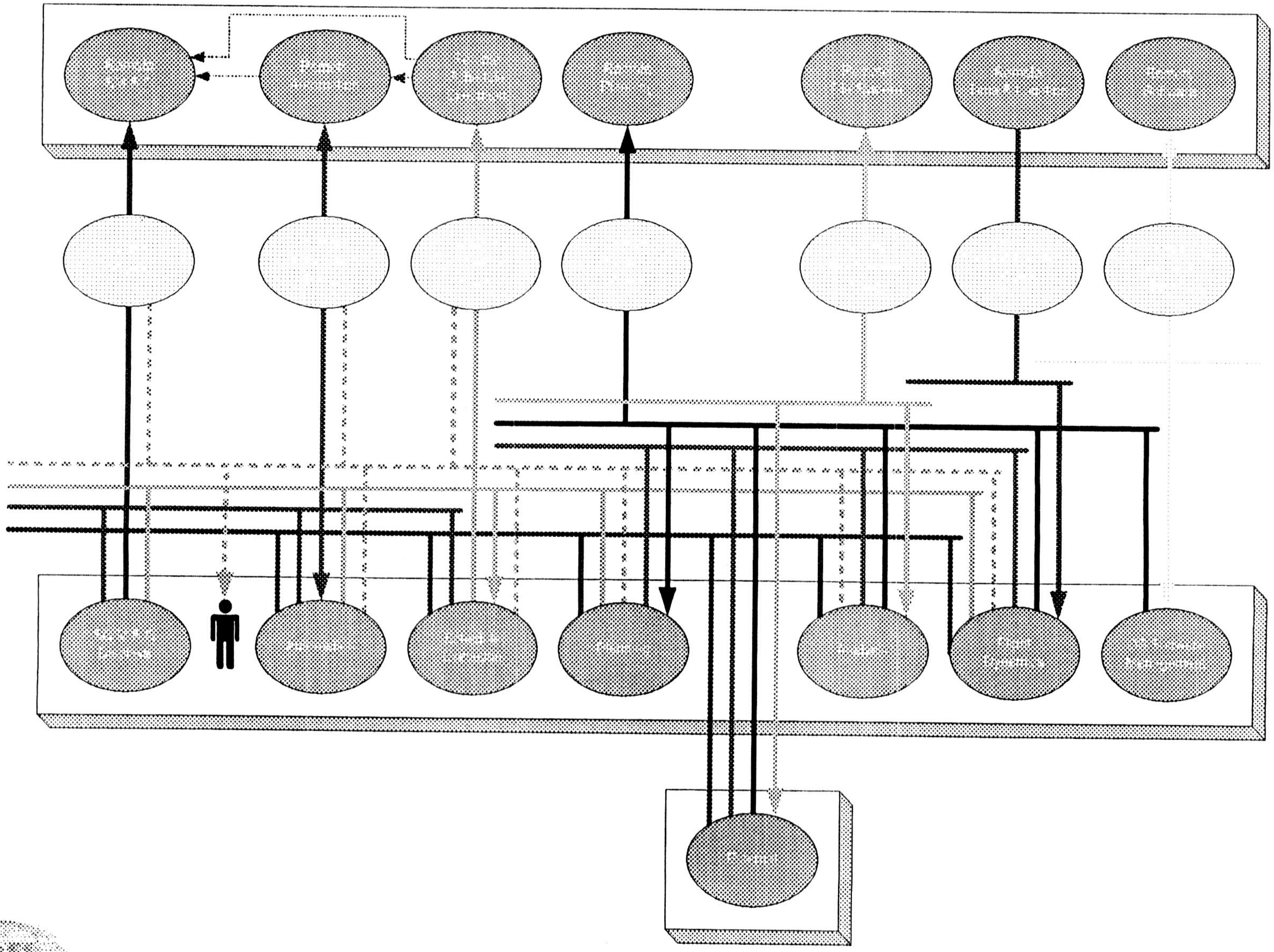
- Key task is to identify the information objects
- Elaboration of the services will identify these objects
- Will also define their attributes



Information flow



Service Overview

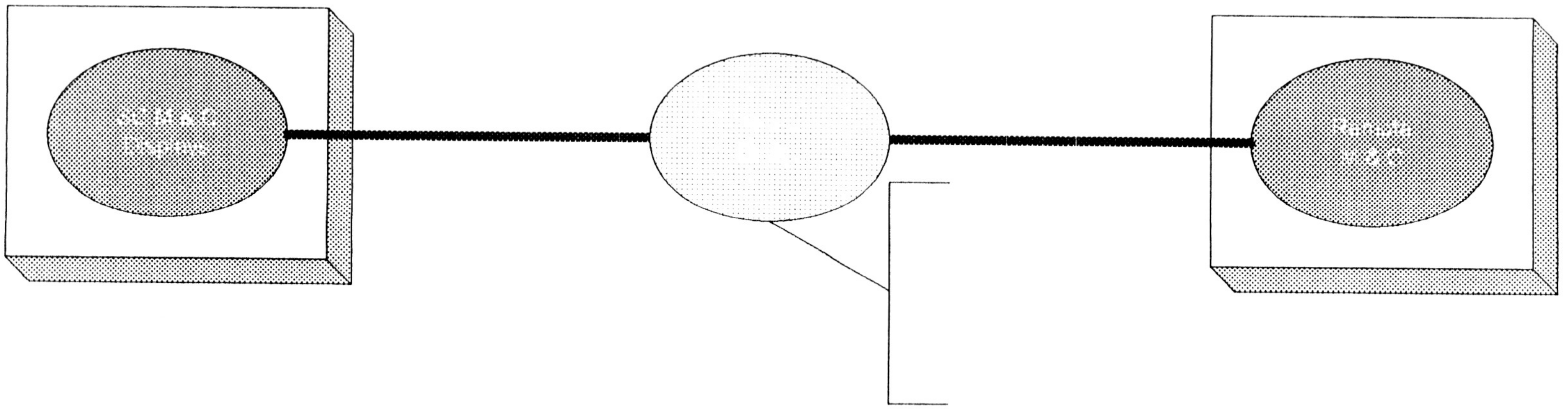


Space Ops 2004

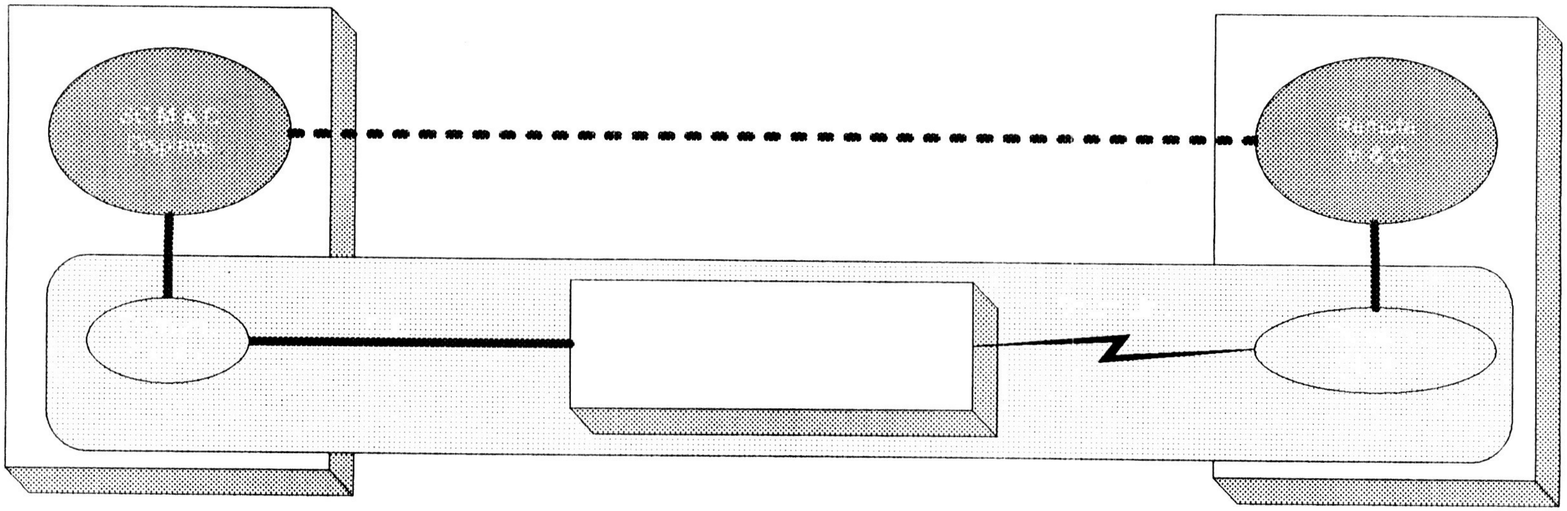
List of services

- M&C
  - Core Monitoring and Control service.
- Automation
  - Activity automation management.
- Scheduling
  - Activity scheduling.
- Interaction
  - Operator notification and interaction.
- Planning
  - Constraint and resource planning.
- Flight Dynamics
  - Orbit determination, flight plan generation, and manoeuvre generation.
- Time
  - Time correlation
- Location
  - Tracking, ranging, and onboard position determination services.
- File Management
  - File management and transfer, both ground based and onboard.
- Software Management
  - Software versioning, patching, and release. Ground based and onboard.

Example: Core SM&C



Example: Core SM&C



Summary

- ◆ Produce CCSDS White Book
  - Issue date is May 2004
- ◆ Analysis of operational context
- ◆ Use of services and layers
- ◆ Identification of services
- ◆ Will lead onto definition of services
  - In other CCSDS books

Any questions?